

22p.

125

N64-19940

CODE-1

NASA CR-53928

Technical Report No. 32-579

*Segmented Rational Minmax Approximation,  
Characteristic Properties and  
Computational Methods*

Charles L. Lawson

OTS PRICE

XEROX

\$

2.60 pk

MICROFILM

\$

0.86 mfe

jpl

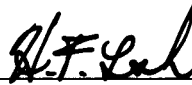
JET PROPULSION LABORATORY  
CALIFORNIA INSTITUTE OF TECHNOLOGY  
PASADENA, CALIFORNIA

December 19, 1963

*Technical Report No. 32-579*

*Segmented Rational Minmax Approximation,  
Characteristic Properties and  
Computational Methods*

*Charles L. Lawson*



---

H. F. Lesh, Chief  
Computer Applications

JET PROPULSION LABORATORY  
CALIFORNIA INSTITUTE OF TECHNOLOGY  
PASADENA, CALIFORNIA

December 19, 1963

Copyright © 1964  
Jet Propulsion Laboratory  
California Institute of Technology

Prepared Under Contract No. NAS 7-100  
National Aeronautics & Space Administration

## CONTENTS

<b>I. Introduction</b>	1
<b>II. Characteristic Properties of the Segmented Rational Minmax Approximation Problem</b>	2
A. Three Basic Lemmas	2
B. The Segmented Rational Minmax Approximation Problem	4
C. Existence of a Balanced Solution Vector	4
D. Sufficiency of the Balanced Error Property	6
E. Possibility of Descent to a Solution	6
F. Counterexamples	7
<b>III. Computation of Segmented Approximations</b>	8
A. Problem I	8
B. Problem II	10
C. Problem III	12
D. Remarks on Continuity of a Segmented Approximator at Breakpoints	14
<b>IV. Conclusions</b>	15
<b>References</b>	16

## FIGURES

1. Illustration of proof of Theorem 3	6
---------------------------------------	---

## TABLES

1. Progress of SEGFIT algorithm in Example 1	9
2. Progress of SEGFIT algorithm in Example 2	10
3. Final approximator computed by SEGFIT algorithm in Example 2	10
4. Progress of SEQFIT algorithm in Example 3	11
5. Final approximator computed by SEQFIT algorithm in Example 3	11
6. Number of segments and parameters resulting from the use of the SEQFIT algorithm in Example 4	12
7. Progress of SEGSQL algorithm in Example 7	13
8. Progress of SEGSQL algorithm in Example 10	14

## FOREWORD

The results of Section II of this Report were presented at the 1962 National Conference of the Association for Computing Machinery. Portions of this Report will appear in *Numerische Mathematik*.

## ABSTRACT

Characterization theorems, solution procedures, and results of numerical examples are reported regarding the problem of partitioning an interval so that the largest error incurred in approximating a continuous function by separate polynomial or rational forms on each subinterval is minimized.

19940 A

Arthur

## I. INTRODUCTION

A problem that must be faced repeatedly in a scientific computing center is that of finding economically computable representations for various known functions. In this context a known function is one for which some usable but possibly uneconomical representation is already available. For example, the function may be defined by a slowly converging power series or by the numerical solution of a differential equation.

Two commonly used methods of function representation that are frequently computationally convenient are table look-up and polynomial approximation. Some generalizations of these methods, which in some cases provide more economy, include rational approximation (i.e., one polynomial divided by another polynomial) and segmented polynomial or segmented rational approximation.

A segmented approximation is one in which different polynomials or rational functions are used for different

subintervals of the argument domain. For some functions, breakpoints for convenient segmentation are suggested by identities satisfied by the functions. In other cases there are no such natural breakpoints and it becomes relevant to pose the following problems:

### Problem I

How can a given interval be partitioned into subintervals so that the maximum error in approximating a given continuous function by  $m$  different polynomials or rational functions of specified degrees will be as small as possible?

### Problem II

What is the smallest integer  $m$  such that a given continuous function can be approximated to within a specified error tolerance on a specified interval by using  $m$  different polynomials or rational functions of specified degrees?

The theory of these approximation problems has previously been studied for some restricted cases of particular interest (Ref. 1-4). Section II gives existence and characterization theorems for Problem I and establishes other results that are useful in suggesting solution methods.

In practice, approximation problems of the type under consideration can vary widely with regard to the number of parameters to be determined and the precision required. Appropriately, therefore, methods of varying degrees of generality have been devised for solving these problems (Ref. 1-3, 5-11). In Section III a computation

method that appears to have a wide range of applicability is described and compared with some of the other reported methods.

A number of computer programs were written in support of the work reported here. Besides the double-precision (54-bit) programs NODFIT, SEGFIT, and SEQFIT, and the single-precision program SEGSQ, two extended-precision arithmetic packages were adapted to FORTRAN to provide 70-bit and 140-bit arithmetic. Versions of NODFIT and SEGFIT were written to use these packages for computing high-precision approximators for various elementary functions.

## II. CHARACTERISTIC PROPERTIES OF THE SEGMENTED RATIONAL MINMAX APPROXIMATION PROBLEM

A rational minmax approximator for a continuous real valued function on a closed, bounded real interval exhibits a characteristic balancing of the extremes of the error curve (Ref. 12). This property has been exploited in some of the methods that have been devised for the numerical solution of the rational minmax approximation problem (Ref. 13-17). Such methods strive iteratively to improve the balance of the extremes of the error curve.

In this Section, it is shown that there is also a property of balanced maximum errors associated with the segmented rational minmax problem. It is a sufficient, but not a necessary, condition for a solution. The segmented problem need not have a unique solution, but it always has some solution that has the balanced error property.

Numerical solution methods for the segmented minmax problem can be based upon this property. Section III describes such a method and gives some numerical examples.

Section IIA is devoted to three independent lemmas that identify the properties of rational minmax approximation and segmentation upon which the remainder of the development rests. Some other settings of practical interest in which Lemmas 1 and 2, and thus all the results of this Report, hold are positively weighted rational minmax approximation and minmax approximation by linear combinations of a given finite set of continuous functions.

In Section IIB, the segmented rational minmax approximation problem is stated, and the existence of a solution is deduced. In Section IIC, the existence of a solution having the balanced error property is established. In Section IID, some inequalities similar to those known in the linear least maximum problem are established, and the balanced error property is shown to be a sufficient condition for a solution.

In Section IIE, it is shown that, for any initial position of the breakpoints defining the segmentation, there is a continuous transformation of the breakpoints, which permits the maximum error to descend to its minimum value. Section IIF provides some examples illustrating the lack of convexity in this problem.

### A. Three Basic Lemmas

Let  $f$  be a continuous real valued function on the non-degenerate closed, bounded real interval  $[\alpha, \beta]$ . Let  $n$  and  $d$  be nonnegative integers. Let  $G$  be the class of rational functions whose numerators and denominators are polynomials of degrees not exceeding  $n$  and  $d$ , respectively. For real numbers  $v$  and  $w$  satisfying

$$\alpha \leq v \leq w \leq \beta$$

define the minmax error function  $h$  by

$$h(v, w) = \min_{g \in G} \max_{v \leq x \leq w} |f(x) - g(x)|$$

**Lemma 1.**<sup>1</sup> *The function  $h$  is continuous on the compact region of  $vw$ -space defined by  $\alpha \leq v \leq w \leq \beta$ .*

**Proof.** Let  $v$  and  $w$  satisfy  $\alpha \leq v \leq w \leq \beta$ . Suppose  $h$  is not continuous at  $(v, w)$ . Then there exists an  $\epsilon > 0$  and sequences  $\{v_i\}$  and  $\{w_i\}$  with  $\alpha \leq v_i \leq w_i \leq \beta$ ,  $i = 1, 2, \dots$ , such that  $\lim_i v_i = v$ ,  $\lim_i w_i = w$ , and

$$|h(v_i, w_i) - h(v, w)| > \epsilon \text{ for all } i = 1, 2, \dots, \quad (1)$$

Let  $g$  (respectively  $g_i$ ,  $i = 1, \dots$ ) denote a member of  $G$  that is the minmax approximator for  $f$  on  $[v, w]$  (respectively  $[v_i, w_i]$ ,  $i = 1, 2, \dots$ ), normalized so that the coefficient of largest magnitude in  $g$  (respectively  $g_i$ ) is 1. An existence theorem for these minmax approximators is given in Ref. 12.

By definition  $|f(x) - g(x)|$  is bounded by  $h(v, w)$  for  $x \in [v, w]$ . Consequently the continuity of  $f$  and  $g$  permits us to choose a  $\delta > 0$  such that

$$|f(x) - g(x)| < h(v, w) + \epsilon$$

for  $x$  in the closed interval  $I$  between  $\max\{v - \delta, \alpha\}$  and  $\min\{w + \delta, \beta\}$ . Without loss of generality, we will assume the points  $v_i$  and  $w_i$ ,  $i = 1, 2, \dots$ , lie in the closed interval  $I$ .

The definitions of  $h$  and  $\delta$  assure that

$$h(v_i, w_i) \leq \max_{v_i \leq x \leq w_i} |f(x) - g(x)| \leq h(v, w) + \epsilon$$

Along with inequality (1) this implies

$$h(v_i, w_i) < h(v, w) - \epsilon \quad i = 1, 2, \dots \quad (2)$$

If  $v = w$ , then  $h(v, w) = 0$ , in which case inequality (2) is impossible and Lemma 1 is established. We proceed to complete the proof for the case  $v < w$ .

The normalization of the rational forms  $g_1, g_2, \dots$ , assures that there is a subsequence of  $\{g_i\}$  whose corresponding coefficients form convergent sequences. Without loss of generality we will assume the sequence  $\{g_i\}$  has this property. Let  $g^*$  denote the rational form whose coefficients are respectively the limits of the sequences of corresponding coefficients in the sequence  $\{g_i\}$ . The

normalization assures that the coefficient of largest magnitude in  $g^*$  is 1.

Let  $x'$  be any point satisfying  $v < x' < w$ . Then for all sufficiently large  $i$  the point  $x'$  also satisfies  $v_i < x' < w_i$ , whence

$$|f(x') - g_i(x')| \leq h(v_i, w_i) \leq h(v, w) - \epsilon \quad (3)$$

for sufficiently large  $i$ . It follows that, unless  $x'$  is a zero of its denominator,  $g^*$  satisfies

$$|f(x') - g^*(x')| \leq h(v, w) - \epsilon.$$

Since  $x'$  was chosen arbitrarily in  $(v, w)$  this relation holds for all  $x$  in  $(v, w)$  except zeros of the denominator of  $g^*$ .

Any point  $x$  in  $(v, w)$  that is an isolated zero of the denominator of  $g^*$  must also be a zero of the numerator of  $g^*$  with at least as great multiplicity, for otherwise inequality (3) would be violated in a neighborhood of  $x$  for sufficiently large  $i$ . Furthermore the denominator of  $g^*$  is not identically zero, for then inequality (3) would require that the same be true of the numerator of  $g^*$ , contradicting the statement that one of the coefficients of  $g^*$  is 1. It follows that if  $g'$  is the rational form obtained from  $g^*$  by removing all polynomial factors common to the numerator and denominator of  $g^*$ , then  $g'$  satisfies

$$|f(x) - g'(x)| \leq h(v, w) - \epsilon \text{ for all } x \in (v, w)$$

Under these circumstances  $g'$  cannot have a pole at  $v$  or  $w$  and thus this bound for  $|f(x) - g'(x)|$  is uniform throughout the closed interval  $[v, w]$ .

This implies that  $g'$  is a better approximator for  $f$  on  $[v, w]$  than the best approximator,  $g$ , whose maximum error is  $h(v, w)$ . This contradiction followed from the assumption that  $h$  was not continuous at  $(v, w)$ . This completes the proof of Lemma 1.

**Lemma 2.** *The minmax error function  $h$  as defined preceding Lemma 1 is nonincreasing in its first variable and nondecreasing in its second variable.*

**Proof.** Let  $v_1 \leq v_0 \leq w_0 \leq w_1$ . Let  $g_1$  be the least maximum approximator in  $G$  for  $f$  on  $[v_1, w_1]$ .

<sup>1</sup>This lemma could be deduced from a more general continuity theorem given in Ref. 18, which in turn depends upon results established in Ref. 19.



Then

$$h(v_1, w_1) = \max_{v_1 \leq x \leq w_1} |f(x) - g_1(x)| \geq \max_{v_0 \leq x \leq w_0} |f(x) - g_1(x)|$$

$$\geq \min_{g \in G} \max_{v_0 \leq x \leq w_0} |f(x) - g(x)| = h(v_0, w_0)$$

**Lemma 3.** Let  $m$  be an integer exceeding 1 and let  $u_i$  and  $v_i$  be numbers satisfying

$$\alpha \equiv u_0 \leq u_1 \leq \dots \leq u_{m-1} \leq u_m \equiv \beta$$

and

$$\alpha \equiv v_0 \leq v_1 \leq \dots \leq v_{m-1} \leq v_m \equiv \beta$$

Then, unless  $u_i = v_i$  for all  $i$ , there exist indices  $j$  and  $k$  such that the following proper inclusions hold:

$$[u_{j-1}, u_j] \subset [v_{j-1}, v_j]$$

and

$$[v_{k-1}, v_k] \subset [u_{k-1}, u_k]$$

**Proof.** Suppose  $u_i \neq v_i$  for some  $i$ . Let  $s$  be the first index for which inequality holds and without loss of generality assume  $u_s < v_s$ . Then the lemma is established by letting  $j = s$  and letting  $k$  be the first index greater than  $j$  for which  $u_k \geq v_k$ .

### B. The Segmented Rational Minmax Approximation Problem

Let  $f$  be a continuous function on  $[\alpha, \beta]$  as in Section IIA. Let an integer  $m \geq 2$  specify the number of contiguous subintervals into which  $[\alpha, \beta]$  is to be partitioned by the selection of  $m - 1$  breakpoints  $u_i, i = 1, \dots, m - 1$ , satisfying

$$\alpha \equiv u_0 \leq u_1 \leq u_2 \leq \dots \leq u_{m-1} \leq u_m \equiv \beta \quad (4)$$

Let  $n_i$  and  $d_i, i = 1, \dots, m$ , be nonnegative integers and let  $G_i$  denote the set of rational functions whose numerators and denominators are respectively polynomials of degrees not exceeding  $n_i$  and  $d_i$ .

For  $i = 1, \dots, m$ , and for  $u_{i-1}$  and  $u_i$  satisfying  $\alpha \leq u_{i-1} \leq u_i \leq \beta$ , define the minmax error function for the  $i$ th subinterval by

$$h_i(u_{i-1}, u_i) = \min_{g_i \in G_i} \max_{u_{i-1} \leq x \leq u_i} |f(x) - g_i(x)|$$

By Lemma 1 each of these functions  $h_i$  is continuous on its domain of definition.

Let  $U$  denote the subset of  $(m+1)$ -space consisting of those vectors  $u = (u_0, u_1, \dots, u_m)$  whose components satisfy condition (4). On the set  $U$  define the maxminmax function  $\mu$  by

$$\mu(u) = \max \{h_i(u_{i-1}, u_i) : i = 1, \dots, m\}$$

Our problem is to minimize  $\mu$  over  $U$ .

The continuity of the functions  $h_i$  implies the continuity of  $\mu$ . The existence of a solution vector  $u^*$  is then an immediate consequence of the fact that  $U$  is compact.

For our later use we introduce the following definitions:

$$\tau = \min \{\mu(u) : u \in U\}$$

$$U^* = \{u : \mu(u) = \tau\}$$

$$\nu(u) = \min \{h_i(u_{i-1}, u_i) : i = 1, \dots, m\}$$

A vector  $u$  will be called balanced if

$$h_i(u_{i-1}, u_i) = \mu(u) \quad i = 1, \dots, m$$

Note that  $\tau$  can be called the *minmaxminmax* error for the problem. It will be shown that  $\tau = \max \{\nu(u) : u \in U\}$  and thus  $\tau$  also deserves the title of *maxminminmax* error.

### C. Existence of a Balanced Solution Vector

At this point it will be useful to introduce a closely related dynamic programming problem.

Define:

$$e_1(u_1) = h_1(\alpha, u_1)$$

and

$$e_i(u_i) = \min_{\alpha \leq u_{i-1} \leq u_i} \max \{e_{i-1}(u_{i-1}), h_i(u_{i-1}, u_i)\}$$

$$i = 2, \dots, m \quad (5)$$

To relate this to the problem formulated in Section IIB note that  $e_i(u_i)$  is the minmaxminmax error for the  $i$ -segment problem on the interval  $[\alpha, u_i]$ . In particular

$e_m(\beta) = \tau$ . Equation (5) represents the fact that an  $i$ -segment minmaxminmax approximator on  $[\alpha, u_i]$  may be found by searching on the single variable  $u_{i-1}$  for the most favorable combination of an  $(i-1)$ -segment minmaxminmax approximator on  $[\alpha, u_{i-1}]$  and a one-segment minmax approximator on  $[u_{i-1}, u_i]$ .

**Theorem 1.** *Statements A, C, and D are valid for  $1 \leq i \leq m$  and statement B for  $2 \leq i \leq m$ .*

- A.  $e_i(\alpha) = 0$
- B. *Given  $v_i \in [\alpha, \beta]$ , there exists  $v_{i-1} \in [\alpha, v_i]$  such that  $e_i(v_i) = e_{i-1}(v_{i-1}) = h_i(v_{i-1}, v_i)$*
- C.  $e_i$  is nondecreasing on  $[\alpha, \beta]$
- D.  $e_i$  is continuous on  $[\alpha, \beta]$

An immediate consequence of statement B in Theorem 1 is the following theorem, which, for the case  $i = m$ , asserts the existence of a balanced solution vector.

**Theorem 2.** *Given  $v_i \in [\alpha, \beta]$ , there exist  $v_j, j = 1, \dots, i-1$ , such that  $\alpha \equiv v_0 \leq v_1 \leq \dots \leq v_i$  and  $e_i(v_i) = h_j(v_{j-1}, v_j), j = 1, \dots, i$ .*

**Proof of Theorem 1.** Statement A, for  $i = 1, \dots, m$ , follows directly from the fact that  $h_i(\alpha, \alpha) = 0, i = 1, \dots, m$ . Statements C and D are valid for  $i = 1$  due to Lemmas 2 and 1, respectively.

Proof will now be given for statements B, C, and D for  $i > 1$  under the induction hypothesis that C and D are valid for  $i-1$ .

To prove B let  $v_i \in [\alpha, \beta]$  be given. On the interval  $\alpha \leq u_{i-1} \leq v_i$ , the function  $e_{i-1}(u_{i-1})$  is nondecreasing and vanishes at the left end, whereas  $h_i(u_{i-1}, v_i)$ , considered as a function of  $u_{i-1}$  only, is nonincreasing and vanishes at the right end. Since both  $e_{i-1}$  and  $h_i$  are continuous, there must be a point  $v_{i-1}$  (not necessarily unique) in  $[\alpha, v_i]$  at which  $e_{i-1}(v_{i-1}) = h_i(v_{i-1}, v_i)$ . Such a point obviously provides the minimum value, among  $u_{i-1} \in [\alpha, v_i]$ , of  $\max \{e_{i-1}(u_{i-1}), h_i(u_{i-1}, v_i)\}$ . This establishes statement B.

For later use we note that the point  $v_{i-1}$  also maximizes the value of  $\min \{e_{i-1}(u_{i-1}), h_i(u_{i-1}, v_i)\}$  among  $u_{i-1} \in [\alpha, v_i]$ . This permits an alternative definition of  $e_i$  for  $i > 1$ :

$$e_i(u_i) = \max_{\alpha \leq u_{i-1} \leq u_i} \min \{e_{i-1}(u_{i-1}), h_i(u_{i-1}, u_i)\} \quad (6)$$

To prove C let  $v_i \leq w_i$  be given. Using B, there exists  $v_{i-1} \in [\alpha, v_i]$  such that  $e_i(v_i) = e_{i-1}(v_{i-1}) = h_i(v_{i-1}, v_i)$ . Then for  $u_{i-1} \in [\alpha, v_{i-1}]$

$$e_i(v_i) = h_i(v_{i-1}, v_i) \leq h_i(u_{i-1}, v_i) \leq h_i(u_{i-1}, w_i) \leq \max \{e_{i-1}(u_{i-1}), h_i(u_{i-1}, w_i)\}$$

while for  $u_{i-1} \in [v_{i-1}, w_i]$

$$e_i(v_i) = e_{i-1}(v_{i-1}) \leq e_{i-1}(u_{i-1}) \leq \max \{e_{i-1}(u_{i-1}), h_i(u_{i-1}, w_i)\}$$

Thus for all  $u_{i-1} \in [\alpha, w_i]$

$$e_i(v_i) \leq \max \{e_{i-1}(u_{i-1}), h_i(u_{i-1}, w_i)\}$$

and so  $e_i(v_i) \leq e_i(w_i)$ , which establishes statement C.

For the proof of D let  $v_i \in [\alpha, \beta]$  and  $\epsilon > 0$ . Using the monotonicity established in C, it will suffice to prove the existence of a  $\delta > 0$  such that, if  $v_i \neq \beta$ ,

$$v_i < u_i < v_i + \delta \text{ implies } e_i(u_i) < e_i(v_i) + \epsilon$$

and, if  $v_i \neq \alpha$ ,

$$v_i > u_i > v_i - \delta \text{ implies } e_i(u_i) > e_i(v_i) - \epsilon$$

By statement B there exists  $v_{i-1} \in [\alpha, v_i]$  such that  $e_i(v_i) = e_{i-1}(v_{i-1}) = h_i(v_{i-1}, v_i)$ . By the continuity of  $h_i$  (Lemma 1), if  $\delta > 0$  is sufficiently small, then  $|h_i(v_{i-1}, u_i) - h_i(v_{i-1}, v_i)| < \epsilon$  for  $|u_i - v_i| < \delta$ .

For  $v_i \neq \beta$  we may assume  $\delta$  is smaller than  $\beta - v_i$ . Then for  $v_i < u_i < v_i + \delta$

$$e_i(u_i) \leq \max \{e_{i-1}(v_{i-1}), h_i(v_{i-1}, u_i)\} = h_i(v_{i-1}, u_i) < h_i(v_{i-1}, v_i) + \epsilon = e_i(v_i) + \epsilon$$

Similarly for  $v_i \neq \alpha$  we may assume  $\delta < v_i - \alpha$ . Then we wish to consider  $u_i$  satisfying  $v_i > u_i > v_i - \delta$ , but two cases arise, depending on whether  $v_{i-1} = v_i$  or  $v_{i-1} < v_i$ . In the first case we have  $e_i(v_i) = h_i(v_{i-1}, v_i) = 0$  and thus, by C,  $e_i(u_i) = 0$  for all  $u_i \in [\alpha, v_i]$ . Thus we certainly have  $e_i(u_i) \geq e_i(v_i) - \epsilon$ .

In the second case we may assume  $\delta$  is smaller than  $v_i - v_{i-1}$  so that  $v_{i-1}$  lies in  $[\alpha, u_i]$ . Then using the alternative definition (6) for  $e_i$  we obtain

$$e_i(u_i) \geq \min \{e_{i-1}(v_{i-1}), h_i(v_{i-1}, u_i)\} = h_i(v_{i-1}, u_i) > h_i(v_{i-1}, v_i) - \epsilon$$

This completes the proof of statement D and hence of Theorem 1.

### D. Sufficiency of the Balanced Error Property

In Section IIC the existence of a balanced solution vector was established. In this Section it will be shown that every balanced vector is a solution vector. We continue to use the notation introduced in Section IIB.

**Lemma 4.** For any  $u$  and  $v$  in  $U$

$$\nu(u) \equiv \min_i h_i(u_{i-1}, u_i) \leq \max_i h_i(v_{i-1}, v_i) \equiv \mu(v)$$

**Proof.** If  $u = v$ , the lemma is trivially true. Suppose  $u \neq v$ . Then by Lemma 3 there is an index  $j$  such that  $[u_{j-1}, u_j] \subset [v_{j-1}, v_j]$ . Therefore,

$$\nu(u) \leq h_j(u_{j-1}, u_j) \leq h_j(v_{j-1}, v_j) \leq \mu(v)$$

where the center inequality is due to Lemma 2.

**Lemma 5.** For any  $u \in U$ ,  $\nu(u) \leq \tau \leq \mu(u)$ .

**Proof.** The second inequality follows directly from the definition of  $\tau$ . To establish the first inequality let  $u^*$  be a solution vector and apply Lemma 4, obtaining

$$\nu(u) \leq \mu(u^*) = \tau$$

An immediate consequence of Lemma 5 is

**Lemma 6.** A balanced vector is a solution vector.

### E. Possibility of Descent to a Solution

**Theorem 3.** For any  $v \in U$  and  $v^* \in U^*$  there is a piecewise linear path  $P$  in  $U$  connecting  $v$  to  $v^*$  and having the property that  $\mu(u)$  is nonincreasing as  $u$  moves along  $P$  from  $v$  to  $v^*$ .

**Proof.** A vector-valued function  $u$  will be defined that maps a real interval  $0 \leq t \leq k$  (for some  $k < m$ ) onto a subset  $P$  of  $U$  in such a way that  $P$  has the properties stated in Theorem 3.

Step 1. Set  $t_c = 0$ . Set  $u_i(0) = v_i$ ,  $i = 0, 1, \dots, m$ .

Step 2. If  $u_i(t_c) = v_i^*$ ,  $i = 0, \dots, m$ , then set  $k = t_c$  and quit; otherwise go to Step 3.

Step 3. Let  $j$  be the first index such that  $[u_{j-1}(t_c), u_j(t_c)]$  is a proper subset of  $[v_{j-1}^*, v_j^*]$ .

We will call  $[u_{j-1}(t), u_j(t)]$  the key variable interval for the current iteration and  $[v_{j-1}^*, v_j^*]$  the key target interval. The effect of Steps 4a and 5a which follow will be to expand the key variable interval so that it coincides with the key target interval when  $t$  reaches  $t_c + 1$ . Steps

4b and 5b provide for components in the path of the expansion to be carried along rather than being bypassed.

Step 4a. For  $t_c < t < t_c + 1$ , let  $u_{j-1}(t)$  vary linearly, taking the value  $v_{j-1}^*$  at  $t_c + 1$ .

Step 4b. If an index  $i < j-1$  satisfies

$$u_i(t_c) \in [v_{j-1}^*, u_{j-1}(t_c)]$$

then, since the value  $u_{j-1}(t)$  varies from the right to the left end of this interval as  $t$  varies from  $t_c$  to  $t_c + 1$ , there must be a point  $t_i$  at which  $u_{j-1}(t_i) = u_i(t_c)$ . Define  $u_i(t)$  to be constant for  $t_c \leq t \leq t_i$  and to be equal to  $u_{j-1}(t)$  for  $t_i \leq t \leq t_c + 1$ .

Step 5a. For  $t_c < t < t_c + 1$ , let  $u_j(t)$  vary linearly, taking the value  $v_j^*$  at  $t_c + 1$ .

Step 5b. If an index  $i > j$  satisfies  $u_i(t_c) \in [u_j(t_c), v_j^*]$ , then there must be a point  $t_i$  at which  $u_j(t_i) = u_i(t_c)$ . Define  $u_i(t)$  to be constant for  $t_c \leq t \leq t_i$  and to be equal to  $u_j(t)$  for  $t_i \leq t \leq t_c + 1$ .

Step 6. For each index  $i$  not treated in Steps 4 or 5, define  $u_i(t)$  to be constant for  $t_c < t < t_c + 1$ .

Step 7. Replace  $t_c$  by  $t_c + 1$  and return to Step 2.

**Remark 1.** The graph in Fig. 1 illustrates a set of functions  $u_i(t)$  defined by the above construction. In this illustration the key variable interval for the first iteration, i.e., as  $t$  varies from 0 to 1, is  $[u_0(t), u_1(t)]$ . The succeeding key variable intervals are  $[u_3(t), u_4(t)]$  and  $[u_2(t), u_3(t)]$ . Note that  $u_2(t)$  coincides with  $u_3(t)$  in the latter part of the interval  $1 \leq t \leq 2$ .

**Remark 2.** The existence of the index  $j$  needed in Step 3 is assured by Lemma 3.

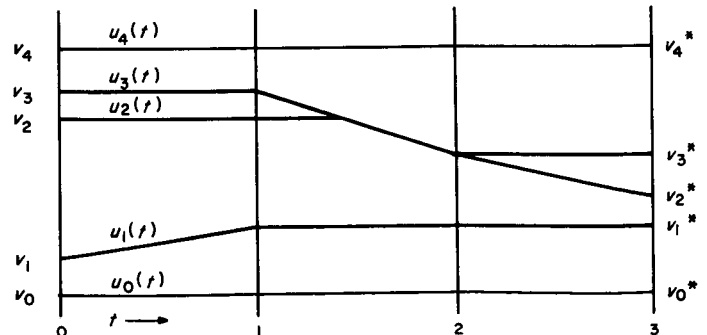


Fig. 1. Illustration of proof of Theorem 3

**Remark 3.** In Steps 4a and 5a at least one of the components  $u_{j-1}$  or  $u_j$  is not equal to its final value ( $v_{j-1}^*$  or  $v_j^*$ , respectively) at  $t = t_c$ , but both are equal to their final values at  $t = t_c + 1$ . Both of these two components remain constant for  $t \geq t_c + 1$ . Thus at least one previously unstabilized component stabilizes on each iteration and so the procedure terminates after, at most,  $m - 1$  iterations.

**Remark 4.** The function  $\mu(u)$  can increase with increasing  $t$  only if one of the functions  $h_i(u_{i-1}, u_i)$  increases as  $t$  increases. For  $h_i(u_{i-1}, u_i)$  to increase it is necessary that either  $u_{i-1}$  decrease or  $u_i$  increase and that  $u_{i-1}$  be distinct from  $u_i$ .

In the given construction only the key variable interval is permitted to move in this manner. Since it is covered by its target interval during the move, it follows from Lemma 2 that  $h_j(u_{j-1}, u_j) \leq h_j(v_{j-1}^*, v_j^*) \leq \tau$ . This move cannot cause an increase in  $\mu$  because  $\mu$  is never less than  $\tau$ . Thus  $\mu$  is nonincreasing as  $t$  goes from 0 to  $k$ ; i.e., as  $u$  goes from  $v$  to  $v^*$  in the prescribed manner.

The set  $\{u(t): 0 \leq t \leq k\}$  has, therefore, all of the properties required of the path  $P$ . This concludes the proof of Theorem 3.

## F. Counterexamples

In the light of the favorable descent properties stated in Theorem 3 it is natural to inquire whether  $\mu$  is a convex function on  $U$ . Example A below shows that this need not be the case. Example B shows that the solution set  $U^*$  can fail to be convex, although Theorem 3 does imply that  $U^*$  is arcwise connected.

**Example A.** This example shows that the function  $\mu$  can fail to be convex and that  $\mu$  can have weak local minima which are not global minima. Define  $f(x)$  for  $|x| \leq 7$  by linear interpolation in the following tabulation:

$x$	-7	-5	-3	-1	1	3	5	7
$f(x)$	-3	-3	-1	-1	1	1	3	3

Consider the problem of approximating  $f$  by two constant functions, i.e.,  $m = 2$ ,  $n_1 = n_2 = d_1 = d_2 = 0$ . The space  $U$  of possible breakpoint vectors consists of all vectors of the form  $(-7, u_1, 7)$  with  $-7 \leq u_1 \leq 7$ . The functions  $h_1$ ,  $h_2$ , and  $\mu$  are then given by linear interpolation in the following tabulation:

$u_1$	-7	-5	-3	-1	0	1	3	5	7
$h_1(-7, u_1)$	0	0	1	1	1.5	2	2	3	3
$h_2(u_1, 7)$	3	3	2	2	1.5	1	1	0	0
$\mu(-7, u_1, 7)$	3	3	2	2	1.5	2	2	3	3

The function  $\mu$  is seen to be nonconvex. If  $f$  is redefined in the interval  $[-1, 1]$  to be  $\sin x\pi/2$ , then the new  $\mu$  will be nonconvex in every neighborhood of the solution vector  $(-7, 0, 7)$ .

The function  $\mu$  has many weak local minima; for example, every point in the open interval between  $u_1 = -3$  and  $u_1 = -1$ . The only strong minimum is the global minimum at  $u_1 = 0$ .

**Example B.** This example shows that  $U^*$  can fail to be convex. Define  $f(x)$  for  $|x| \leq 4$  by linear interpolation in the following tabulation:

$x$	0	$\pm 1$	$\pm 2$	$\pm 3$	$\pm 4$
$f(x)$	-1	2	1	4	3

We will approximate  $f$  by three linear polynomials, i.e.,  $m = 3$ ,  $n_1 = n_2 = n_3 = 1$ ,  $d_1 = d_2 = d_3 = 0$ . One solution is  $u = (-4, -2, 0, 4)$ ,  $g_1(x) = -x$ ,  $g_2(x) = -x$ ,  $g_3(x) = x$ ,  $\mu = 1$ . Another solution is  $v = (-4, 0, 2, 4)$ ,  $g_1(x) = -x$ ,  $g_2(x) = x$ ,  $g_3(x) = x$ ,  $\mu = 1$ . The midpoint between  $u$  and  $v$ , namely  $(-4, -1, 1, 4)$ , does not provide a solution however, since the best approximator on  $[-1, 1]$  is  $g_2(x) \equiv 0.5$  and this permits a maximum error of 1.5.

### III. COMPUTATION OF SEGMENTED APPROXIMATIONS

Piecewise linear approximations have found application in the design of diode function generators (Ref. 3) and in the linearization of nonlinear constraints in mathematical programming (Ref. 6). Segmented functions using polynomials of higher degree and rational forms have been used in function generation subroutines for digital computers (Ref. 1, 5, 11, 20).

Three different types of segmented approximation problems will be defined, with a suggested computation method for each. These methods have been programmed and some numerical examples will be reported.

#### A. Problem I

In this formulation of the segmented approximation problem, there is given a continuous function  $f$  on a closed interval  $[\alpha, \beta]$ , an integer  $m$  specifying the number of subintervals into which  $[\alpha, \beta]$  is to be partitioned, and integers  $n_i$  and  $d_i$ ,  $i = 1, \dots, m$ , specifying the degrees of the numerator and denominator, respectively, of the rational form to be used in approximating  $f$  on the  $i$ th subinterval. The problem is to determine  $m - 1$  breakpoints  $u_i$ ,  $i = 1, \dots, m - 1$  satisfying

$$\alpha \equiv u_0 \leq u_1 \leq u_2 \leq \dots \leq u_{m-1} \leq u_m \equiv \beta \quad (7)$$

and rational functions  $r_i$  of the specified degrees such that

$$\max_{i=1, \dots, m} \max_{u_{i-1} \leq x \leq u_i} |f(x) - r_i(x)|$$

is minimized.

If it is assumed that a method is available (Ref. 13-17) for computing a least maximum approximator  $r_i$  once a subinterval  $[u_{i-1}, u_i]$  is specified, then the problem may be viewed as the minimization of a real valued function of  $m - 1$  variables,  $u_i$ ,  $i = 1, \dots, m - 1$  subject to the constraint (7). This minimization problem can be given a dynamic programming formulation (Section IIC and Ref. 8-10).

In designing a solution method, it is helpful to have more information about the structure of the problem. In particular, it was shown in Section II that the maximum absolute error  $h_i$  of the least maximum approximator on the  $i$ th subinterval depends continuously upon the endpoints of the subinterval. There always exists a set of breakpoints that produces the condition  $h_1 = h_2 = \dots = h_m$

and such a breakpoint set is a solution. Given any breakpoint set, there exists a continuous transformation of the breakpoints that moves them to a position at which  $h_1 = h_2 = \dots = h_m$  in such a way that  $\max_i h_i$  is nonincreasing throughout the transformation.

Let  $\tau$  denote the minimum value of  $\max_i h_i$ . Upper and lower bounds on  $\tau$  are provided respectively by the largest and smallest of the  $h_i$ 's associated with any particular set of breakpoints.

The results stated above suggest that an iterative solution method could be based upon an attempt to balance the  $h_i$ 's by shortening the subintervals having the larger  $h_i$ 's and lengthening the subintervals having the smaller  $h_i$ 's. The details of such an algorithm will depend upon what assumption is made regarding the dependence of  $h_i$  upon the breakpoints  $u_{i-1}$  and  $u_i$ .

One such algorithm will be discussed, one that has been programmed and successfully used to obtain segmented approximations for some of the elementary functions and some other functions of practical interest. We assume that, for small changes in  $u_{i-1}$  and  $u_i$ , the minmax error  $h_i$  depends only upon the length,  $s_i = u_i - u_{i-1}$ , of the  $i$ th subinterval. We further assume that the dependence is of the form  $h_i = k_i s_i^{a_i}$  where  $k_i$  is an unknown positive constant and  $a_i = n_i + d_i + 1$ . For typographical convenience, we define  $c_i = 1/a_i$ .

The choice of the form  $h = k s^a$  was motivated by the results established by Maehly and Witzgall (Ref. 18, 19; also see 1, 21) for small  $s$ . Computational studies involving rational and polynomial fits for the sine and exponential function on various intervals indicate that this form provides a useful guide to the minmax error behavior over a useful range of values of  $s$ .

Suppose now that some choice of breakpoints has been made and let  $s_i$  and  $h_i$ ,  $i = 1, \dots, m$ , denote the associated subinterval lengths and minmax errors respectively. We seek new breakpoints and a number  $H$  such that the error on each of the new subintervals will have the common value  $H$ . The number  $H$  and the lengths  $S_i$  of the new subintervals are related by the following  $m + 1$  equations:

$$\frac{H}{S_i^{a_i}} = \frac{h_i}{s_i^{a_i}} \quad i = 1, \dots, m$$

$$\sum_{i=1}^m S_i = \beta - \alpha$$

A single equation involving only the unknown  $H$  can be obtained as follows:

Define  $b_i = s_i/h_i^{c_i}$ ,  $i = 1, \dots, m$ . Then

$$S_i = b_i H^{c_i} \quad i = 1, \dots, m \quad (8)$$

and

$$\sum b_i H^{c_i} - (\beta - \alpha) = 0 \quad (9)$$

If all  $a_i$ 's have a common value  $a$  then equation (9) admits an explicit solution

$$H = \left[ \frac{(\beta - \alpha)}{\sum_i b_i} \right]^a$$

If the  $a_i$ 's are not all equal then Newton's method may be used. For equation (9) the iteration formula is

$$H_{(j+1)} = H_{(j)} \left[ 1 - \frac{\sum_i b_i H_{(j)}^{c_i} - (\beta - \alpha)}{\sum_i c_i b_i H_{(j)}^{c_i}} \right]$$

The Newton iteration proceeds without difficulty since the left member of equation (9) is monotone in  $H$  for  $H \geq 0$ . Since  $H$  must lie between  $\min_i h_i$  and  $\max_i h_i$ , some number in that interval can be used to start the iteration.

After determining  $H$ , the  $S_i$ 's may be computed directly from equation (8). This procedure has been implemented

in a FORTRAN subroutine called SEGFIT. The subroutine which SEGFIT uses to compute a least maximum rational approximator on a single subinterval is called NODFIT and uses Maehly's second direct method (Ref. 13, 14).

**Example 1.** Ream (Ref. 3, 7) reported an interesting noniterative technique for computing segmented linear polynomial nearly minmax approximations that requires the integration of  $|f''|^{1/2}$ . One of his examples was a five-segment linear approximation for  $e^{-x}$  on  $[0, 1]$ . Table 1 summarizes four iterations of the program SEGFIT applied to this problem. The breakpoints listed under Pass 4 are the same as those given by Ream and the errors are also in agreement.

**Example 2.** A two-segment rational approximator for  $\sin x\pi/2$  on  $[0, 1]$  that could be used in a 15-decimal-place floating-point subroutine for sine and cosine was obtained by using SEGFIT to find the least-maximum-relative-error fit using the form  $xS_i(x^2)/S_1(x^2)$  on the first segment and  $C_5((1-x)^2)/C_1((1-x)^2)$  on the second segment. Here  $S_i$  and  $C_i$  denote polynomials of degree  $i$ . When fitting with odd or even functions, we use  $a_i = n_i + d_i + 2$  rather than  $a_i = n_i + d_i + 1$ . Table 2 summarizes the progress of this computation and Table 3 lists the coefficients of the final approximator.

Note, in Table 2, that the number of NODFIT iterations per SEGFIT pass dropped considerably after Pass 1.

Table 1. Progress of SEGFIT algorithm in Example 1

	Pass 1		Pass 2		Pass 3		Pass 4	
$u_0$	0.0		0.0000		0.0000		0.0000	
$h_1$		0.00227		0.001516		0.001547		0.0015488
$u_1$	0.2		0.1621		0.1639		0.1639	
$h_2$		0.00185		0.001561		0.001546		0.0015486
$u_2$	0.4		0.3413		0.3423		0.3425	
$h_3$		0.00152		0.001580		0.001548		0.0015485
$u_3$	0.6		0.5393		0.5383		0.5386	
$h_4$		0.00124		0.001567		0.001551		0.0015485
$u_4$	0.8		0.7581		0.7559		0.7560	
$h_5$		0.00102		0.001520		0.001550		0.0015487
$u_5$	1.0		1.0000		1.0000		1.0000	
$H$		0.001488		0.001548		0.0015488		0.0015486

**Table 2. Progress of SEGFIT algorithm in Example 2**

	Pass 1	Pass 2	Pass 3	Pass 4	Pass 5
$u_1$	0.5	0.4173	0.415493	0.4154475	0.4154465
$h_1$	1.888 (6)	0.213 (2)	0.20186 (2)	0.201597 (1)	0.2015888 (1)
$h_2$	0.021 (9)	0.192 (3)	0.20135 (2)	0.201585 (2)	0.2015905 (1)
$h_1-h_2$	1.867	0.021	0.00051	0.000012	-0.0000017
$H$	0.180	0.20103	0.201574	0.2015905	0.2015897

The fixed endpoints are  $u_0 = 0$ ,  $u_2 = 1$ . The data  $h_1$ ,  $h_2$ ,  $h_1-h_2$  and  $H$  are given in units of  $10^{-15}$ . The number of NODFIT iterations used to compute each  $h_i$  is given in parentheses below each value of  $h_i$ . Total running time = 2 min 50 sec on IBM 7090 computer.

**Table 3. Final approximator computed by SEGFIT algorithm in Example 2**

$$\sin t\pi/2 \cong t \left( \frac{\sum_{i=0}^4 s_i t^{2i}}{s_5 + t^2} \right), |t| \leq u_1$$

$$\cos t\pi/2 \cong \left( \frac{\sum_{i=0}^5 c_i t^{2i}}{c_6 + t^2} \right), |t| \leq 1-u_1$$

$$u_1 = 0.4154465$$

$$\text{Maximum relative error} = 0.2016 \cdot 10^{-15}$$

$i$	$s_i$				$c_i$			
0	69.787	72489	96816	54013	53.163	31207	42924	07420
1	-27.128	25520	70405	54734	-64.587	60735	31183	04668
2	2.894	63946	94512	58433	12.252	21066	22690	00859
3	-0.128	30950	56392	83336	-0.855	50223	30740	11058
4	0.002	44647	32443	48351	0.028	00745	79761	61843
5	44.428	24553	96867	14143	-0.000	42063	88543	45713
6					53.163	31207	42923	96703

In fact, the total number of NODFIT iterations used in Passes 2 through 5 was about the same as the number used in Pass 1. Since NODFIT accounts for most of the running time, this implies that Passes 2 through 5 together required only about as much time as Pass 1. This efficiency of later passes is the result of implementing the assumption that the relative locations of the zeros of a least maximum residual curve are approximately invariant with respect to small changes of the interval.

The choice of the rational forms used in this example deserves some explanation. For each index  $i = 0, 1, \dots, 5$  we computed the least maximum approximator of the form  $xS_i(x^2)/S_{5-i}(x^2)$  for  $\sin x\pi/2$  on  $[0, 0.5]$ . The minimax

error was found to be a convex function of  $i$  with a minimum at  $i = 4$ .

C. Witzgall, in an unpublished manuscript, has shown that the Padé approximators for sine have similar behavior; i.e., among the six Padé approximators of the form  $xS_i(x^2)/S_{5-i}(x^2)$  the leading term of the error series has smallest magnitude when  $i = 4$ . This is of particular interest since it has sometimes been assumed that the diagonal or near-diagonal (here  $i = 2$  or 3) forms of the Padé table have the smallest error.

## B. Problem II

In this formulation of the segmented least maximum approximation problem, we assume that an acceptable error  $\tau$  is specified and the number  $m$  of segments is to be determined. If the error is related to the subinterval length by the simple formula  $h_i = k_i s_i^{a_i}$ , then the following algorithm suggests itself:

Step 1. Guess a value of  $s_1$ .

Step 2. Use some minmax approximation method to compute  $h_1$ .

Step 3. Obtain a new estimate  $S_1$  by computing

$$S_1 = \left( \frac{\tau}{h_1} \right)^{c_1} s_1$$

Step 4. Replace  $s_1$  by  $S_1$  and return to Step 2.

When  $s_1$  has been determined to acceptable accuracy, the procedure can be repeated for  $s_2$ , etc.

A variation of this procedure would be to compute a new value for  $a_i$  after each iteration on the  $i$ th segment rather than using a preset value of  $a_i$  throughout. This amounts to *regula falsi* iteration using  $\log s$  and  $\log (h/\tau)$  as the independent and dependent variables, respectively.

This variation could also be used in Problems I and III (Section IIC), but the only program in which it has been incorporated is SEQFIT, which was written to solve Problem II.

**Example 3.** The program SEQFIT was used to compute a sequential segmented minmax approximator for sine  $x$  using the ratio of two quadratic polynomials in the variable  $t_i = x - c_i$  as the approximating form on the  $i$ th subinterval, where  $c_i$  is the abscissa of the center of the  $i$ th subinterval. The acceptable error was specified as  $\tau = 0.0005$  and a relative tolerance of 0.005 was permitted in attaining this value. The program was permitted to run until five segments has been determined.

The action of the subroutine NODFIT, which was used by SEQFIT to obtain the least maximum rational approximator on each subinterval specified by SEQFIT, was terminated when it had leveled the residual curve to within a relative tolerance of 0.0005 or after 11 iterations if this tolerance was not met.

When NODFIT executes  $k$  iterations it solves  $2k-1$ , or slightly fewer,  $n$  by  $n$  systems of linear equations (5 by 5 in this example), and evaluates the object function (here

sine) about  $12k(n+1)$  to  $25k(n+1)$  times. In searching for peaks of the residual curve a tolerance of  $0.0001s'$  was permitted in the abscissa of a peak where  $s'$  is the distance between the interpolation nodes bracketing that peak. Relaxing this tolerance would reduce the number of function evaluations per iteration.

The progress of the computation is summarized in Table 4, and the final breakpoints and coefficients are given in Table 5. This approximator is not proposed as a useful approximator for sine, but was computed for the purpose of observing the behavior of SEQFIT in approximating an oscillating function.

Table 4. Progress of SEQFIT algorithm in Example 3

SEQFIT iteration number	Subinterval number	Subinterval length in units of $\pi$	Number of NODFIT iterations Total = 88	Maximum residual in units of $10^{-4}$
1	1	0.2223	2	0.056
2		0.546	3	2.84
3		0.622	3	4.46
4		0.643	3	4.95
5		0.645	2	4.9993
6	2	0.645	4	13.05
7		0.532	3	4.63
8		0.540	2	5.007
9	3	0.540	11	0.59 <sup>a</sup>
10		0.828	9	10.05
11		0.745	11	3.82 <sup>a</sup>
12		0.7671	11	5.04 <sup>a</sup>
13		0.7665	6	4.997
14	4	0.767	4	11.12
15		0.653	3	6.81
16		0.591	2	4.76
17		0.599	2	5.009
18	5	0.599	3	7.24
19		0.556	2	4.68
20		0.563	2	5.0003
*Unreliable; NODFIT did not level the residual curve to within the relative tolerance 0.0005.				

Example 4. In certain applications of an 85-foot azimuth-elevation-mounted microwave antenna the positioning of the antenna is controlled via paper tape, which carries the desired settings of azimuth and elevation at 64-second increments of time. This tape is prepared by a large computer at a location remote from the antenna.

In support of a study of alternative control techniques, the program SEQFIT was used to obtain segmented polynomial minmax approximators to the data that would normally have been supplied to control the antenna in observing the circumpolar radio source Cassiopeia A for a particular 24-hour period.

The desired minmax error was specified as  $\tau = 0.001$  degrees of arc. Table 6 shows the number of segments required for each fit and the total number of parameters needed to specify each segmented approximator. Note, for instance, that the azimuth is fully specified by 43 parameters (five internal breakpoints, two endpoints, and 36 coefficients) when the segmented quintic approximator is used as compared with the 1350 values needed when the azimuth is tabulated every 64 seconds.

Table 5. Final approximator computed by SEQFIT algorithm in Example 3

For $x \in [u_{i-1}, u_i]$ , define $c = (u_{i-1} + u_i)/2$ , $t = x - c$ , then $ \sin x - (a_{0i} + a_{1i}t + a_{2i}t^2)/(1 + b_{1i}t + b_{2i}t^2)  \leq 0.0005009$						
$i$	$u_i$	$a_{0i}$	$a_{1i}$	$a_{2i}$	$b_{1i}$	$b_{2i}$
0	0.0					
1	2.025661	0.848423	0.452584	-0.380803	-0.093591	0.111229
2	3.721576	0.264771	-0.948520	-0.154654	0.071152	0.175283
3	6.129165	-0.977729	0.173391	0.405009	0.040619	0.097956
4	8.010714	0.708095	0.628713	-0.343935	-0.113236	0.128285
5	9.777817	0.505975	-0.809599	-0.271415	0.110422	0.152417



**Table 6. Number of segments and parameters resulting from the use of the SEQFIT algorithm in Example 4**

Function		Quadratic	Cubic	Quartic	Quintic
Hour angle	Segments	6	4	4	
	Parameters	25	21	25	
Declination	Segments	7	5	5	
	Parameters	29	26	31	
Azimuth	Segments	37	14	8	6
	Parameters	149	71	49	43
Elevation	Segments	31	11	8	6
	Parameters	125	56	49	43
Total IBM 7090 machine time: 54 min					

### C. Problem III

Another segmented approximation problem that has received some attention in the literature is the least squares problem. Here the given data could take the same form as in Problem I, but the object function to be minimized is

$$\sum_{i=1}^m \int_{u_{i-1}}^{u_i} [f(x) - r_i(x)]^2 dx$$

Although programs exist that are used to solve the rational least squares problem, the present discussion will be restricted to segmented polynomial least squares approximation.

Even with this restriction there is a complication not present in Problem I. The object function, regarded as a function of the breakpoints, can have strong local minima that are not global minima.

This situation is illustrated by Example 5, and Example 6 shows the possibility of a disconnected solution set.

**Example 5.** Compute a two-segment zeroth-order approximator for  $x^2$  on the interval  $[-1, 1.2]$ . Explicitly, the problem is to determine numbers  $b$ ,  $c_1$ , and  $c_2$  to minimize

$$\int_{-1}^b (x^2 - c_1)^2 dx + \int_b^{1.2} (x^2 - c_2)^2 dx$$

Since  $c_1$  and  $c_2$  can be written as explicit functions of  $b$ , the above object function can be written explicitly as a function of the single variable  $b$  and in fact turns out to be simply a fourth-degree polynomial in  $b$ :

$$(1/9)[(1+a)b^4 - (1-a^2)b^3 - (1+a^3)b^2 + (1-a^4)b^1 - (1+a^5)] + (1/5)(1+a^5)$$

where  $a$  denotes the right endpoint of the interval; in this example,  $a = 1.2$ .

On the interval  $[-1, 1.2]$  this polynomial attains an absolute maximum value of 0.3218 at  $b = -1$ ,  $-0.20$ , and  $1.2$ , a local minimum of 0.2857 at  $b = 0.76$ , and an absolute minimum of 0.1458 at  $b = 0.81$ .

**Example 6.** Consider the problem of Example 5 on the symmetric interval  $[-1, 1]$ . A minimum value of 0.1222 occurs at  $b = \pm 0.71$  and a maximum value of 0.1778 occurs at  $b = \pm 1$  and  $b = 0$ .

The existence of such examples as these shows the need for more study of Problem III toward the goal of discovering conditions under which a local minimum may be recognized as being a global minimum.

### Solution Methods for Problem III

In spite of the pitfalls inherent in this problem, cases of practical interest have been encountered and apparently solved.

Ream (Ref. 3, 7) gave a noniterative method for obtaining an approximate solution to this problem when the functions  $r_i$  are linear polynomials. His procedure requires the integration of  $|f''|^{2/5}$ .

Stone (Ref. 6) also solved this problem for the case in which the  $r_i$ 's were linear polynomials. He treated all of the variables uniformly, applying Newton's method to the system of nonlinear equations obtained by setting all of the partial derivatives equal to zero.

Bellman (Ref. 8) and Gluss (Ref. 9) regarded the problem as being primarily a search for the correct  $u_i$ 's with the understanding that for any choice of  $u_i$ 's the least squares approximators  $r_i$  can be computed. A significant feature of this approach is the reduction of the number of variables whose values must be sought simultaneously by a nonlinear method. In Ref. 8, 9 the search for the  $u_i$ 's is formulated as a dynamic program.

As in Problem I, the design of an efficient solution method will be aided by consideration of the special properties of the problem at hand. Let  $e_i^-$  (respectively  $e_i^+$ ) denote the magnitude of the difference at the left (respectively right) end of the  $i$ th subinterval between  $f$  and its least squares approximator on that subinterval. Then the equations expressing the nullity of the

partial derivatives of the object function with respect to the  $u_i$ 's can be written as

$$e_i^+ = e_{i+1}^- \quad i = 1, \dots, m-1 \quad (10)$$

At this point we need some assumption regarding the dependence of  $e_i^\pm$  upon  $u_i$  and  $u_{i-1}$ . We propose to assume that  $e_i^\pm = k_i^\pm s_i^{a_i}$ , where  $k_i^\pm$ ,  $s_i$ , and  $a_i$  have the same meaning as in Problem I.

Suppose now that the numbers  $s_i$ ,  $e_i^-$ , and  $e_i^+$ ,  $i = 1, \dots, m$ , are associated with some choice of breakpoints and we seek new breakpoints so that the associated subinterval lengths  $S_i$  and endpoint errors  $E_i^\pm$  will satisfy the  $m$  equations

$$E_{i+1}^- = E_i^+ \quad i = 1, \dots, m-1 \quad (11)$$

$$\sum_{i=1}^m S_i = \beta - \alpha \quad (12)$$

The numbers  $k_i^\pm$  can be computed as  $k_i^\pm = e_i^\pm / s_i^{a_i}$ . Then assuming  $E_i^\pm = k_i^\pm S_i^{a_i}$ , equation (11) becomes

$$k_{i+1}^- S_{i+1}^{a_{i+1}} = k_i^+ S_i^{a_i} \quad i = 1, \dots, m-1$$

These equations permit the expression of each  $S_i$  in terms of  $S_1$ . To this end define  $b_1 = 1$  and

$$b_i = \frac{b_{i-1} k_{i-1}^+}{k_i^-} \quad i = 2, \dots, m$$

Then we obtain

$$S_i^{a_i} = b_i S_1^{a_1} \quad i = 1, \dots, m$$

or

$$S_i = (b_i S_1^{a_1})^{1/a_i} \quad i = 1, \dots, m \quad (13)$$

Replacing  $S_i$  in equation (12) by the right member of (13) provides a single equation involving the single unknown  $S_1$ . This equation may be solved explicitly for  $S_1$  if all  $a_i$ 's have a common value and otherwise may be solved by Newton's iteration. With  $S_1$  known, the other  $S_i$ 's may be computed directly by equation (13).

This procedure has not been programmed in full generality; however, a program SEGSQ was written to handle approximation by segmented linear polynomials in order to obtain a comparison with the cases reported by Stone and Ream.

**Example 7.** The function  $e^{-x}$  is to be approximated over  $0 \leq x \leq 3$  by a two-segment linear approximator. There are thus just one internal breakpoint  $u_1$  and four coefficients to be determined. As is shown in Table 7, the

**Table 7. Progress of SEGSQ algorithm in Example 7**

	Pass 1	Pass 2	Pass 3	Pass 4
$e_1^+$	0.080	0.0509	0.05206	0.052008
$u_1$	1.5	1.0622	1.0799	1.0791
$e_2^-$	0.024	0.0535	0.05195	0.052013
$e_2^- - e_1^+$	-0.056	+0.0026	-0.00011	+0.000005
rms	0.0521	0.0380	0.0380	0.0380

value of  $u_1$ , which was started at 1.5, was changed to 1.0622 after the first pass and the rms of 0.0380 remained unchanged in its first three significant digits thereafter. The difference ( $e_2^- - e_1^+$ ), which the algorithm attempts to force to zero, exhibits linear convergence to zero with a ratio of about 0.045.

The final pair of linear approximators obtained was  $0.9355-0.6001x$  and  $0.4416-0.1425x$ . This agrees with Stone's results to the three figures published. His breakpoint was  $u_1 = 1.080$ .

Ream (Ref. 3) estimated  $u_1 = 1.074$  and  $\text{rms} = 0.0260$  using his noniterative approximate method. Actually computing the least-squares approximators using Ream's value for  $u_1$  we obtained  $\text{rms} = 0.0380$ . This computation was checked using a desk calculator and 0.0380 is correct to the figures given.

**Examples 8 and 9.** The problem of Example 7 was run using three and four segments, starting in each case with equally spaced breakpoints. In Example 8, the rms descended as follows: 0.0248, 0.0169, 0.0168, 0.0168, and in Example 9: 0.0143, 0.0095, 0.0094, 0.0094. The difference from Stone's results was at most 0.008 in the breakpoints and 0.003 in the coefficients.

**Example 10.** The problem of Example 7 was run using 15 segments. Note that this requires the determination of 14 breakpoints and 30 coefficients. It would require higher precision than the 27-bit arithmetic that was used to accurately compute the rms in this case, but the convergence can be judged by noting the apparently linear convergence toward zero of the quantities ( $e_{i+1}^- - e_i^+$ ). Representative data is given in Table 8. The lengths of the final subintervals varied monotonically from  $s_1 = 0.1193$  to  $s_{15} = 0.3593$ . The largest change in any breakpoint from Pass 4 to Pass 5 was 0.0016.

Here the breakpoints were determined to within about  $\pm 0.0016$ , and the computation involved the solution of

Table 8. Progress of SEGSEQ algorithm in Example 10

	Pass 1	Pass 2	Pass 3	Pass 4	Pass 5
$e_1^+$	0.00296	0.000913	0.001096	0.0011058	0.0011029
$e_2^-$	0.00252	0.000981	0.001085	0.0011084	0.0011008
$(e_2^- - e_1^+)$	-0.00044	+0.000068	-0.000011	+0.0000026	-0.0000021
$e_7^+$	0.00089	0.001025	0.0009197	0.0009232	0.00093036
$e_8^-$	0.00076	0.001042	0.0009246	0.0009242	0.00093055
$(e_8^- - e_7^+)$	-0.00013	+0.000017	+0.0000049	+0.0000010	+0.00000019
$e_{14}^+$	0.000219	0.000604	0.000684	0.00066870	0.0006654
$e_{15}^-$	0.000186	0.000552	0.000672	0.00066953	0.0006664
$(e_{15}^- - e_{14}^+)$	-0.000033	-0.000052	-0.000012	+0.00000083	+0.00000010
rms	0.0011	0.0007	0.0007	0.0007	0.0007

the basic one-segment approximation problem 75 times with very little additional computing. In comparison, note that a straightforward application of Newton's method would require setting up and solving a 44 by 44 system of linear equations for each iteration.

In a straightforward computation based on dynamic programming, one might replace the interval  $[0, 3]$  by a grid of 61 equally spaced abscissas (grid spacing = 0.05). The computation would then entail the construction of 14 tables of about 60 entries each and require the solution of the basic one-segment approximation problem about 23,000 times.

#### D. Remarks on Continuity of a Segmented Approximator at Breakpoints

Among the additional conditions that might be imposed in a segmented approximation problem, one of the first that comes to mind is the requirement of continuity of the approximator at the breakpoints. This Report does not consider computational methods for imposing this constraint<sup>2</sup>, but in this Section certain cases are identified in which continuity does not actually constitute an additional constraint.

In view of equation (10) the magnitude of the discontinuity of a segmented least squares approximator at

the breakpoint  $u_k$  is either  $2e_k^+$  or 0. In the least maximum case, the jump could be any magnitude from 0 to  $2\tau$ . In cases such as Examples 7, 8, 9, and 10, where a convex function is being approximated by segmented linear polynomials, the discontinuity is clearly zero. This special case is thoroughly treated in Ref. 2.

This situation of zero discontinuity occurs in segmented  $n$ th degree polynomial approximation whenever  $n$  is odd and the function being approximated has a continuous nonvanishing  $(n+1)$ st derivative throughout the entire interval. This follows from the fact that in  $n$ th degree polynomial least squares or least maximum approximation, the residual curve must have at least  $n+1$  distinct interior zeros and, due to the nonvanishing  $(n+1)$ st derivative, at most  $n+1$  zeros.

If  $z_{ij}$  ( $j = 0, \dots, n$ ) denote the zeros of the residual curve on the  $i$ th subinterval then the residual curve admits the representation

$$\frac{(x - z_{i,0}) \cdots (x - z_{i,n}) f^{(n+1)}(\xi)}{(n+1)!}$$

and therefore, since  $n+1$  is even, agrees in sign with  $f^{(n+1)}$  at the breakpoints. As to the magnitude of the errors at breakpoints, equality for the least squares case is a consequence of equation (10), while in the least maximum case the nonvanishing  $(n+1)$ st derivative implies that the endpoints of each subinterval are points

<sup>2</sup>This constraint is treated in Ref. 2 and 9 and useful relevant ideas can be gleaned from the literature on spline curve fitting; e.g., Ref. 4, 22, 23.

at which the residual curve reaches its maximum magnitude of  $\tau$ .

Another situation in which continuity at the breakpoints need not be considered as a special constraint is that in which the approximator is intended to be used by a processor whose sensitivity is characterized by a nonzero "just noticeable difference" (jnd). Relative to such a processor the approximator will be effectively continuous if it has no discontinuities larger than the jnd.

Thus if a segmented least maximum approximator can be obtained with  $\tau$  less than half the jnd, it will be effectively continuous.

It may be that a discontinuity of specified magnitude  $\alpha$  is tolerable at the breakpoints, but it is too costly, in some sense, to require  $\tau$  to be less than  $\alpha/2$ . Then weights may be introduced, having increasing magnitude near the breakpoints, in order to force the error, and thus the discontinuity, to be smaller at the breakpoints.

#### IV. CONCLUSIONS

Problems I and III fall into the general class of parametric minimization problems and as such may be attacked, and very likely solved, by a wide range of general-purpose procedures, such as Newton's method, gradient or other descent methods, or various algorithms based on the recursive relations of a dynamic programming formulation. In applications in which the number of parameters to be computed is sufficiently small, and the precision requirement is sufficiently low, it probably makes little difference what method is used. In more difficult problems, the difference in efficiency of the various methods is more significant.

A study of Problem I resulted in the finding that the solution set (in the breakpoint-vector space) is connected and that from any starting point there is a descent path to the solution set. With this information available, some descent methods were tried, of which the procedure reported in Section IIA was found to be most satisfactory.

The procedure developed for Problem I was then adapted to Problems II and III. It should be kept in mind, however, that, in view of Example 5, a descent procedure for Problem III can lead to a local minimum that is not an absolute minimum.

The procedures presented in this Report appear to be more efficient for the class of problems considered than the straightforward use of more general methods.

A more general dynamic programming approach would be useful in cases in which (1) the data are essentially discrete (Ref. 10), (2) additional constraints are to be satisfied (Ref. 9, 24), or (3) in Problem III when a more complete scan of the solution space is desired to improve the probability that the computed local minimum is indeed a global minimum.

#### ACKNOWLEDGMENT

The author wishes to acknowledge the assistance of N. Block, C. Coltharp, W. Jackson, D. Leistico, C. Moler, and W. Rand in writing the computer programs that supported this work.

## REFERENCES

1. Harrison, J. O., Jr., "Piecewise Polynomial Approximations for Large-Scale Digital Calculations," *Mathematical Tables and Other Aids to Computation*, Vol. 3, 1949, pp. 400-407.
2. Remes, E. Ya., *General Computational Methods of Chebyshev Approximation: The Problems with Linear Real Parameters*, A. E. C. Translation No. 4491, original, 1957, translation, 1962, pp. 296-313.
3. Ream, N., "Approximation Errors in Diode Function-Generators," *Journal of Electronics and Control*, Vol. 7, 1959, pp. 83-96.
4. Johnson, R. S., "On Monosplines of Least Deviation," *Transactions of the American Mathematical Society*, Vol. 96, 1960, pp. 458-477.
5. Muller, Mervin E., "An Inverse Method for the Generation of Random Normal Deviates on Large Scale Computers," *Mathematical Tables and Other Aids to Computation*, Vol. 12, 1958, pp. 167-174.
6. Stone, H., "Approximation of Curves by Line Segments," *Mathematics of Computation*, Vol. 15, 1961, pp. 40-47.
7. Ream, N., "Note on 'Approximation of Curves by Line Segments,'" *Mathematics of Computation*, Vol. 15, 1961, pp. 418, 419.
8. Bellman, R., "On the Approximation of Curves by Line Segments Using Dynamic Programming," *Communications of the Association for Computing Machinery*, Vol. 4, June 1961, p. 284.
9. Gluss, B., "Further Remarks on Line Segment Curve-Fitting Using Dynamic Programming," *Communications of the Association for Computing Machinery*, Vol. 5, August 1961, pp. 441-443.
10. Fryer, W. D., *Best Approximation in Chebyshev Sense of N Line Segments to a Curve by Means of Dynamic Programming*, paper presented November 2, 1962, at the Fall Meeting of the Society for Industrial and Applied Mathematics.
11. Fraser, W., and J. F. Hart, "Near-Minimax Polynomial Approximations and Partitioning of Intervals" (to be published).
12. Achieser, N. I., *Theory of Approximation*, Frederick Ungar Publishing Co., New York, 1956, p. 55.
13. Maehly, H., "Rational Approximations for Transcendental Functions," in *Information Processing*, Oldenbourg, Munich, 1960, pp. 57-62.
14. Maehly, H., "Methods for Fitting Rational Approximations, Parts II and III," *Journal of the Association for Computing Machinery*, Vol. 10, 1963, pp. 257-277.
15. Cheney, E. W., and H. L. Loeb, "Two New Algorithms for Rational Approximation," *Numerische Mathematik*, Vol. 3, 1961, pp. 72, 75.
16. Cheney, E. W., and T. H. Southard, "A Survey of Methods for Rational Approximation, with Particular Reference to a New Method Based on a Formula of Darboux," *Society for Industrial and Applied Mathematics Review*, Vol. 5, 1963, pp. 219-231.

## REFERENCES (Cont'd)

17. Fraser, W., and J. F. Hart, "On the Computation of Rational Approximations to Continuous Functions," *Communications of the Association for Computing Machinery*, Vol. 5, 1962, pp. 401-403, and 414.
18. Maehly, H., and C. Witzgall, "Tschebyscheff-Approximationen in kleinen Intervallen II. Stetigkeitssätze für gebrochen rationale Approximationen," *Numerische Mathematik*, Vol. 2, 1960, pp. 293-307.
19. Maehly, H., and C. Witzgall, "Tschebyscheff-Approximationen in kleinen Intervallen I. Approximation durch Polynome," *Numerische Mathematik*, Vol. 2, 1960, pp. 143-159.
20. Kogbetliantz, E. G., "Generation of Elementary Functions," in *Mathematical Methods for Digital Computers*, John Wiley & Sons, Inc., New York, 1960, pp. 7-35.
21. Nitsche, Johannes C. C., "Über die Abhängigkeit der Tschebyscheffschen Approximierenden einer differenzierbaren Funktion vom Intervall," *Numerische Mathematik*, Vol. 4, 1962, pp. 262-276.
22. Theilheimer, F., and W. Starkweather, "The Fairing of Ship Lines on a High-Speed Computer," *Mathematical Tables and Other Aids to Computation*, Vol. 15, 1961, pp. 338-355.
23. Ahlberg, J. H., and E. N. Nilson, "Convergence Properties of the Spline Fit," *Journal of the Society for Industrial and Applied Mathematics*, Vol. 11, 1963, pp. 95-104.
24. Fryer, W. D., *Dynamic Programming Solution for the Ideal Low-Level Flight Path, with General Curve-Fitting Implications*, paper presented November 2, 1962, at the Fall Meeting of the Society for Industrial and Applied Mathematics.